Informe de elaboración

Autores: Paula Alejandra Cadena Espitia

Cristian Santiago Ortiz Vargas

Juan Camilo Acosta Rojas

**Funciones:**

1. **Ingresar:**

En Cliente:

**int ingresar(char nombre[32],char tipo[32],int edad,char raza[16],int estatura,float peso, char sexo);**

La función crea una estructura dogType con la información ingresada y la envía al servidor para que este la guarde. Recibe confirmación del ingreso por parte del servidor e imprime la estructura ingresada al cliente.

En Servidor:

int ingresar(struct dogType n,int clientfd);

La función ingresar en el proceso del servidor permite que dicho programa almacene tanto en la tabla hash como en la lista de registros de la veterinaria la estructura a ingresar, esto se logra mediante la recepción de todos los datos necesarios para la creación del registro a ingresar, esto se hace mediante el envío de dichos datos mediante el proceso del cliente, luego de esto envía al proceso cliente el registro ya ingresado para que muestre su correcto ingreso.

1. **Ver registro:**

En cliente:

int verRegistro()

La función verRegistro en el cliente recibe la estructura del registro buscado y verifica si se quiere ver la historia clínica, envía al servidor el mensaje con la respuesta del cliente y entonces recibe el archivo de historia clínica si la hay, lo abre para modificarlo y lo envía de vuelta al servidor si el archivo es guardado

En servidor:

int verRegistro(int id, int clientfd, int confver);

La función verRegistro en servidor permite que mediante el id que se reciba por parte de la aplicación cliente pueda buscar en la lista de registros de la veterinaria uno que tenga el id solicitado y que previamente se haya verificado su existencia en dicha lista de registros, luego de encontrar el registro envía los respectivos datos al proceso cliente para mostrarle los respectivos datos.En caso de que confver sea 1 acaba ahí , sino pedirá confirmación para ver la historia clínica. Y si recibe confirmación de que el cliente quiere ver la historia clínica verifica su existencia en el servidor y en tal caso la envía, si el cliente modifica la historia, se elimina la del servidor y se recibe el archivo modificado por el cliente.

1. **Buscar:**

En Cliente

**int Buscar();**

Recibe del Servidor la cantidad de estructuras cuyo nombre coincide con el nombre buscado. Utiliza una estructura dogType, para recibir las estructuras del Servidor y va imprimiendolas para el usuario una a una.

En servidor:

int Buscar(char nombre[32],int clientfd);

La función buscar en el proceso servidor permite que mediante un nombre el cual es enviado por parte del proceso cliente le envíe a dicho proceso una lista de registros los cuales pertenecen a los registros de la veterinaria, los cuales envía uno por uno para no congestionar el envío de información entre los dos procesos.

1. **Eliminar:**

En Cliente:

**int eliminar();**

Recibe del Servidor la estructura que se desea eliminar, imprima la información de la estructura para el usuario y pide confirmación para continuar con la eliminación, esta decisión es enviada al Servidor para continuar el proceso, y espera confirmación de parte de este de su fin.

En servidor:

int eliminar (int id, int clientefd);

La función eliminar en el proceso servidor permite que mediante un id enviado por el proceso cliente elimine de la lista de registros de la veterinaria el registro identificado con ese id, lo primero a realizar es obtener los datos de dicho registro para mostrarselos al proceso cliente, luego de esto el proceso cliente envía una confirmación para su posterior eliminación, y actualiza el archivo reduciendo su tamaño, aumentando el número de eliminados que va a ser importante para la correcta asignación del id y eliminando el registro del archivo

1. **Listar:**

int Listar(int Hash)

La función Listar sólo está en el servidor, y al recibir un número Hash, va a la tabla hash y a partir de ella genera la lista encadenada de todas las mascotas que tienen el hash ingresado en un apuntador global al cual se le asigna memoria dinámicamente y guarda la cantidad de registros en esa lista en la variable global n

1. **PolyHash:**

**int PolyHash (char cadena[]);**

La función PolyHash en el proceso servidor permite que dado un arreglo de caracteres donde se va a asignar a dicho arreglo el nombre de un registro que se piensa almacenar en la lista de registros de la veterinaria. Esta función le asigna un valor entero a dicha cadena de caracteres para poder asignarle un lugar en la tabla hash de los respectivos registros.

1. **mygetch:**

Solo existe en Cliente

**int mygetch ( void );**

funcion tomada de https://faq.cprogramming.com/cgi-bin/smartfaq.cgi?id=1043284385&answer=1042856625

Utilizada para que el en el menú el cliente espere que se oprima una tecla antes de continuar, lee el input de la consola, desactiva la entrada canónica, lee los caracteres anteriores que no hayan sido leídos y espera el siguiente input, resetea la entrada de la consola y termina